REMARKS

The above Amendments and these Remarks are in reply to the Office Action mailed November 9, 2004. Claims 49-53, 55-61, 63-70 and 72-75 are pending herewith for consideration.

Rejection of Claims 49-75 Under 35 U.S.C. §102(e)

Claims 49 – 75 were rejected under 35 U.S.C. §102(e) as anticipated by U.S. Patent No. 5,710,922 Alley et al. (Alley).

Alley discloses synching a "local" computer to a "remote" computer when the remote computer is docked to the local computer. In particular, as described at Col. 7, line 51 through Col. 10, line 17, in order to synch a remote computer to a local computer, the local computer includes a docking interface allowing the remote computer to be docked thereto. If the local and remote computers are not compatible, then the docking operation is terminated and no synch is possible. (Col. 9, lines 42-46). If the systems are compatible, then the remote computer is synched to the local computer. The synching process is time based. As explained in the reference:

> Initially in step 142, the dock sends a message to the remote which indicates the last time that the remote was synchronized by that particular dock. The time of the last synchronization is sent in order to determine which entries need to be updated. As indicated above, when an entry is revised, the revision time is entered as part of the data entry. This permits the synchronization to be time based. That is, only the records that have been added, deleted or revised since the last synchronization need to be revised. By maintaining a time-based backup system, multiple docks can be used to synchronize the information stored on a particular remote, which may be desirable for a variety of reasons. (Col. 10, lines 19-31).

A. Alley Requires Physical Docking

The Examiner summarized Applicants' first argument as being that "Alley does not teach 'a network for coupling the first file system and second file system to allow communication between the [sic] first file system and the second file system,...' ." While this is true, the Examiner omitted the remaining portion of applicants' argument: that the communication between the first and second file systems occurs, "when the first and second file systems are physically remote from each other." The Examiner did not address this portion of applicants' argument and did not explain where this express claim limitation is found in Alley.

It is respectfully submitted that this claim limitation is not found in Alley. Alley clearly and throughout discloses that the pen-based computer system is synched with the desk top computer only when the pen-based computer is docked to the desk top computer. There is no disclosure, teaching or suggestion in Alley of a system where two computing devices may be synched when they are remote from each other.

Without such disclosure, it is respectfully submitted that Alley cannot form an anticipation of the present invention. If the Examiner maintains the rejection on these grounds, it is respectfully requested that the Examiner specifically point out where these limitations are found in the cited reference.

B. Synchronization by Universal Differencing Data

Applicant also argued in the previous Response to Office action that Alley does not disclose, teach or suggest a system as in the present invention whereby application data is first converted to a universal format, and it is this universal format data that is used for a comparison against a stored prior version of the data. Converting the application data to a universal format overcomes the problem of synching application data that may be in different formats. The Examiner did not address this argument in the current Office action, and in fact Alley has no disclosure teaching or suggestion of a system where application data is first converted to a universal format, and it is this universal format data that is used for a comparison against a stored prior version of the data.

In fact, Alley expressly discloses that where data is incompatible, the synch operation is impossible and is terminated:

When communication between the dock and the host is initiated, the local logic first determines whether the systems are compatible in step 132. If not, a dialog box indicating that the systems are not compatible is displayed in step 133 and the docking function is terminated. This determination is made based on the information that is received from the remote in its original connection request. Specifically, in the original connection request, the remote transmits information indicative of the system protocal version. When this information is received, the dock checks to determine whether it is using the same protocal version as (i.e., is capable of synchronizing with) the selected system. Step 132. If so, the logic proceeds to step 135, where it requests, and step 136, where it receives, the remote name and system information. (Alley, Col. 9, lines 43-57)

As explained above, Alley uses a time based system. All data records are time stamped. Upon docking and initiation of the synchronization process, Alley takes all records which have been changed since the last synchronization process, as indicated by the record time stamp, and transfers those changed records between the computers being synchronized.

By contrast, the synchronization process according to the present invention involves converting all application data into a universal format that is not application specific. The universal data for a device is then compared against a stored version of the information for that device to generate universal differencing information. It is this universal differencing information that is used to synchronize other devices. *See*, applicant's disclosure at page 20, line 7 through page 22, line 2:

Figure 9A illustrates a single device engine utilized with a generic application 810 and a generic storage server 850. Figure 9A illustrates a desktop device engine, since all processing occurs on the device and only difference information is transmitted to server 850. Nevertheless, an understanding of the desktop device engine will aid in understanding server side devices engines, hereinafter described. Shown in Figure 9 are the functional components of a device engine in block form

and their interrelationship to each other. The device engine 860 is equivalent to the functional block of a differencing sequencer 104 shown in Figures 1-7.

While the invention will be described with respect to the embodiment of the invention as a differencing synchronizer 104, it will be readily understood that portions of the functionality are utilized as needed in a forward-only (a differencing transmitter) or a receive-only (a differencing receiver) capacity as required by the particular application.

As noted above, a device engine exists for each and every device that makes up a user's personal information network of devices in the system. As shown in Figure 9A, each device engine 860 includes an application object 910. The application object is specific to each particular application 810 and provides a standard interface between the device engine and the balance of the data transmission system of the invention, and the application 810. Details of the application object will be described in further detail below. The application object is a pluggable architecture which supports a wide variety of vendor-unique applications. The job of the application object is to map data from the application into a temporary or "universal" data structure by connecting to the application via any number of standard interfaces to gain access to the applications data. The data structure of the application object puts the data in a generic or "universal data" format which may be used by the device engine components to generate data packages for provision to the storage server.

Also provided is an application object store (AOS) 920 which includes a copy of the device's data at a point just after the previous data extraction and synchronization occurred. Application object store 920 is a mirrored interface which stores a snapshot of the previous state of the data from the application object 910 in the

device engine. The size of the AOS will depend on the data being collected by each device engine.

The generic output of the application object is provided to a delta module 950. Delta module 950 is a differencing engine which calculates differences in data between the output of the application object 910 and the copy of the data which is provided in an application object store (AOS) 920. The actual differencing and patch routine can comprise a routine such as XDelta or YDelta. The delta module 950 will be referred to herein alternatively in certain portions of the description as "CStructuredDelta." In addition, the difference information is alternatively referred to herein as a "change log." Each change log (or set of difference information) is a self describing series of sync transactions. As described below, the change log may be encrypted and compressed before output to the network.

Hence, during a sync, the Application Object will, using a mechanism discussed below, extract the data of each application in the device and convert it to a universal data format. The delta module will then generate a difference set by comparing the output of the Application Object and the AOS. This difference information is forwarded to the encryption and compression routines for output to the storage server 850 in the form of a data package. Alternatively, the data from one application can be used to synchronize to data in another application in, for example, a windows environment, as shown by arrow 1050 in Figure 10.

This method of using application specific interfaces to convert application specific data to a universal format and comparing that universal data to a prior version of the information is nowhere disclosed, taught or suggested in Alley. As described above, Ally simply takes all records which have changed since the last synch, as indicated by a time stamp, and transfers those records. There is no conversion to a universal format and there is no comparison against a prior version of the

information. In the Office action, the Examiner indicated that Alley performs a comparison of information as claimed. It is respectfully submitted that this is not correct. Alley only examines time stamps to determine whether a record has changed since the last synchronization.

The above-described features and distinctions of the present invention are recited in the claims at least as follows:

- "The data synchronization system of claim 49 wherein each said data synchronizer comprises: a data source interface; [and] a copy of a previous state of each said data source" (Claim 51);

- "The data synchronization system of claim 51 wherein said difference information is transmitted from said first synchronizer to said second synchronizer in a universal format" (Claim 52);

- "The data synchronization system of claim 51 wherein said data synchronizer includes a plurality of difference source interfaces, each corresponding to a data source format." (Claim 53);

- "The method of claim 66 wherein said step of determining comprises: comparing data from the first file to a copy of a previous state of data from the first file" (Claim 67);

- "The method of claim 67 wherein said comparing step comprises extracting data from said first file, converting said data to a universal file format, providing said copy of said data in said universal format, and comparing said data and said copy to provide difference data in said universal format" (Claim 68);

- "The method of claim 68 wherein said step of applying comprises: constructing new file data for said second file in said universal data format" (Claim 69);

- "The method of claim 69 wherein said step of updating comprises translating said new file data into a format of said second file." (Claim 70).

To support a rejection under section 102, the Examiner has the burden of showing that each of the claimed features is found in a single prior art reference that was filed at least as early as the claimed invention. *Rockwell Intern. Corp. v. U.S.*, 147 F.3d 1358, 1363 (Fed. Cir. 1998). Omission of any claimed element, no matter how insubstantial, is grounds for traversing a rejection based on §102. *Connell v. Sears, Roebuck & Co.*, 722 F.2d 1542 (Fed. Cir. 1983). As the cited reference has no disclosure of the above discussed features of the claimed invention, including network coupled devices, and does not attempt to address the problem addressed by the present invention, it is respectfully submitted that the present invention is patentable over the cited reference, and it is respectfully requested that the rejection of Claims 49-53, 55-61, 63-70 and 72-75 on Section 102 grounds be withdrawn.

Based on the above, reconsideration of Claims 49-53, 55-61, 63-70 and 72-75 is respectfully requested.

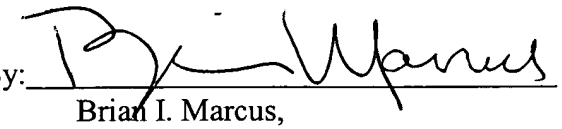
The Examiner's prompt attention to this matter is greatly appreciated. Should further questions remain, the Examiner is invited to contact the undersigned attorney by telephone.

Enclosed is a PETITION FOR EXTENSION OF TIME UNDER 37 C.F.R. § 1.136 for extending the time to respond up to and including today, May 6, 2005.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 501826 for any matter in connection with this response, including any fee for extension of time, which may be required.

Respectfully submitted,

Date: May 6, 2005

By: 

Brian I. Marcus,
Reg. No. 34,511

VIERRA MAGEN MARCUS HARMON & DENIRO LLP
685 Market Street, Suite 540
San Francisco, California 94105-4206
Telephone: (415) 369-9660
Facsimile: (415) 369-9665